### Query Types

Query types of Logpresso are roughly classified into four types depending on how they are executed.

#### Ad-hoc Query

An ad-hoc query refers to arbitrary queries that you can create and execute at any given time. You can programmatically execute queries at any given time through the log query menu of Logpresso's web console, the Logpresso terminal accessed via SSH, or the Logpresso client SDK.

If you switch a query that runs for a long time to the background, it will continue to run even when the current session is logged out or disconnected. You can then switch that query back to the foreground to see the result of the query.

#### Real-time Query

A real-time query refers to a query that processes data received in real time while running for a specified time range from the point of execution. It can use the logs in real time as soon as logs are collected by the log collector, the results of the stream query are output, or the data is provided in a table. This is useful for diagnosis because it immediately samples data without storing all the data on the disk.

Real-time query commands include [logger](https://docs.logpresso.com/en/query/logger-command), [stream](https://docs.logpresso.com/en/query/stream-command), and [table](https://docs.logpresso.com/en/query/table-command).

#### Stream Query

A stream query refers to a query that runs indefinitely in the background for real-time data sources until the system shuts down. A stream query runs queries while continuously guaranteeing the input sequence.

For ENT and STD, you can find stream queries in **Query > Stream**. Stream queries support following three stream types as input:

Log Collector

All logs collected by the logger are provided into the stream query. For ENT and STD, you can configure the logger in **LOG SOURCE**, or for MAE and SNR, in **Collect > Connector**.

Table

Every time a new row is written to the table, it is provided into the stream query. It can be thought of as an evolved use case of the triggers used in a relational database management system (RDBMS). For ENT and STD, you can configure the table in **Table**, or for MAE and SNR, in **System > Table**.

Stream Query

You can use the output of other stream queries as input. Take, for example, a scenario in which a stream query that parses for unstructured logs is placed at the frontend, and multiple analysis stream queries that use that stream query as input are placed. A stream query is divided into streaming mode and refresh mode.

**Streaming mode**

If the query is configured only with commands—streamable commands—that do not rely on data input completion, you can set the stream query in streaming mode.

**Refresh mode**

For example, in the case of statistics and sorting, you can run operations only after the input of the entire data is completed, so the input completion signal is passed at regular intervals.

If you use stream queries to calculate statistics for specific time units and to store the result in an intermediate statistics table, and you design this table to query for the final statistical results, you can get statistical results in real time for large data streams while using very little disk space. You can use Groovy scripting for highly complex real-time analysis and processing.

#### Scheduled Query

Scheduled queries are executed according to the schedule specified by the user. You can selectively save query results and send the results that match the alarm conditions by email.

For ENT and STD, you can load the query results using the **Saved query results** in **Query > Load**.